Subject: Re: FFT and ROTATE
Posted by Wox on Thu, 04 Sep 2008 11:55:27 GMT
View Forum Message <> Reply to Message

On Wed, 3 Sep 2008 15:20:19 -0700 (PDT), wheinz@gmail.com wrote:

> Hello,
>
> I have been wrestling with the FFT and ROTATE functions recently. One
> of the properties of the Fourier transform is that the transform of a
> rotated object is equal to the rotation of the transform of the
> unrotated object. To test this in IDL, I took the FFT of an nxn array
> (called image) and the FFT of that array rotated 90 degrees, image90 =
> ROTATE(image,1). Then, I sorted the real and imaginary parts of the
> coefficients of the results of the FFTs and compared the sorted
> values. I expected that the sorted list of real parts from the FFT of
> the original and rotated arrays would be identical, and that the same
> would be true for the imaginary parts. This is not the case. The sets
> of the magnitudes of the coefficients are equal, as expected.

First of all, care must be taken when rotating the fourier transform. Check IDL help on this: you have to shift with half the image size in both directions. I also noticed that it only works with uneven image sizes (i.e. rotation around the center pixel). Anyway, check the code below:

```
pro rotFFTtest
  ;;load an image
  fn = filepath('md1107g8a.jpg',SUBDIRECTORY='examples/data')
  image=bytarr(251,251)
  image[0,0]= read_image(fn)
  image90 = rotate(image,1)

  ;display the images
  window,0
  tvscl,image,0
  tvscl,image90,1

  n = size(image,/dim)
  nfreq=n/2+1 ; # positive freq in each dim
  nfreq_m=nfreq-1-(~(n mod 2)) ; # negative fequencies in each dim

  ;;take fft of image, then get the real and imaginary parts
  f = fft(image)
  f = shift(f,-nfreq[0],-nfreq[1])
  f90_1 = rotate(f,1)
  f90_1 = shift(f90_1,nfreq[0],nfreq[1])

  ;;take the fft of image90 then get the real and imaginary parts.
  f90_2 = fft(image90)

  tvscl,fft(f90_1,/inverse),2
  tvscl,fft(f90_2,/inverse),3
end
```